UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/477,169	01/04/2000	DONALD STERN	CISCP125	8786

22434 7590 02/16/2005

BEYER WEAVER & THOMAS LLP
P.O. BOX 70250
OAKLAND, CA 94612-0250

EXAMINER	
CAO, DIEM K	

ART UNIT	PAPER NUMBER
2126	

DATE MAILED: 02/16/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/477,169	STERN, DONALD	
	Examiner	Art Unit	
	Diem K Cao	2126	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *09 August 2004*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3-7 and 10-41* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3-7,18-28,34 and 35* is/are rejected.

7)☒ Claim(s) *10-17, 29-33, and 36-41* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1, 3-7 and 10-41 are pending in the application. Applicant has amended claims 1, 19, 22, and 25 and cancelled claim 2.

Continued Examination Under 37 CFR 1.114

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 8/9/2004 has been entered.

Allowable Subject Matter

3. Claims 10-17, 29-33, and 36-41 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1, 3-4, 6-7, and 18-26 are rejected under 35 U.S.C. 103(a) as being unpatentable

over Anderson et al. (U.S. 5,448,735) in view of Munro (Writing DLLs for Windows using Visual Basic, part 1).

6. **As to claim 1**, Anderson teaches determining one or more code modules to be executed (a task is a data ... one or more module; col. 6, line 64 - col. 7, line 32), wherein the one or more code modules are one or more library routines (modules may be either programmed ... library routines, col. 17, lines 26-38), ascertaining a hierarchical order in which the one or more code modules are to be executed (the modules in a task are grouped in the appropriate order; col. 7, lines 6-19), loading the one or more code modules to be executed (load and connect modules in the desired arrangement; col. 7, lines 21-32 and client loads module; col. 9, lines 26-50), and building a chain connecting the one or more code modules such that the one or more code modules will automatically execute in the hierarchical order when a first one of the one or more code modules is executed (A DSP task such as 611 ...by the module; col. 16, line 58 - col. 17, line 46), wherein each of the code modules responsible for calling a next one of the code modules in the chain includes a reference to the next one of the code modules in the chain (the control of execution flow within task may be accomplished by placing references in each module to subsequent modules; col. 19, line 18-30), wherein an address in memory at which the next one of the code modules in the chain is loaded is associated with the reference to the next one of the code modules in the chain (linked using pointers wherein pointer is a variable that contains the address of a location in memory; col. 17, line 62 - col. 18, line 7), wherein the chain does not include a main program (a task is a data structure ... DSP modules; col. 7, lines 6-8, The modules are all related functions... heading; col. 17, lines 4-8, task datum 1201 and its

associated module data ... PhoneMan; col. 17, lines 48-50. Though, Anderson teaches the chain includes only modules which maybe purchase from the off the shelf), and wherein building a chain enables the one or more code modules to execute without requiring a main program responsible for calling the one or more code modules (task datum 1201 ... 1205; col. 17, line 48 - col. 18, line 7).

7. However, Anderson does not teach the one or more code modules are one or more DLLs. Munro teaches DLLs are special libraries that load into memory only once at run-time (page 1).

8. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Anderson and Munro because the advantages of DLLs are well known because DLLs can be loaded only once and used by many applications, and the DLLs can call external functions.

9. **As to claim 3**, Anderson teaches the loading step is performed simultaneous with the building step (DSP modules are ... its function; col. 7, lines 20-32 and to execute module 500; col. 9, lines 25-50).

10. **As to claim 4**, Anderson teaches building a chain is performed such that the one or more code modules can be modified without requiring recompilation of the one or more code modules (The actual executable routine required ... code modules 1202-1205; col. 17, line 67 - col. 18, line 7).

11. **As to claim 6**, Anderson does not explicitly teach determining one or more code modules to be executed comprises determining one or more code modules to be executed to complete configuration of a hardware interface of a router. Anderson teaches determining one or more code modules to be executed to creating tasks for a device (DSP device, real-time task list and share task list; col. 9, line 51 - col. 11, line 7). It would have been obvious to apply the teaching of Anderson to configuration a hardware interface of a router because it provides an efficient means for task organization which groups tasks by functions.

12. **As to claim 7**, see rejection of claim 6 above.

13. **As to claim 18**, Anderson teaches the device is coupled to a bus and resides on the main system logic board (col. 9, line 51-67), and each task has a starting point (Fig. 6) for execution, and associating one of the one or more code modules with a hardware interface to identify a starting point for execution upon occurrence of an interrupt (col. 19, line 65 – col. 20, line 32).

14. **As to claim 19**, it corresponds to the method claim of claim 1. Anderson further teaches a method of configuring a hardware device (create tasks for client and device managers; col. 6, lines 36-61 and configure a phone answering machine task; col. 17, line 48 - col. 18, line 27).

15. **As to claim 20**, see rejection of claim 18 above.

16. **As to claim 21**, see rejection of claim 7 above.

17. **As to computer product claim 22**, it corresponds to the method claim of claim 1.

18. **As to claims 23-24**, see rejections of claims 3-4 above.

19. **As to computer system claim 25**, it corresponds to the method claim of claim 1. Anderson further teaches (col. 4, line 51 - col. 5, line 54) a processor (a processor 102), and a memory (a random access memory).

20. **As to claim 26**, Anderson teaches the address in memory at which the next one of the code modules in the chain is loaded is included in each of the code modules responsive for calling the next one of the code modules in the chain (1202 through 1205 ... linked using pointers ... will be referred to in fields contained with each of the data structure elements; col. 17, lines 48 - col. 18, line 7).

21. Claim 5 is rejected under 35 U.S.C. 103(a) as being unpatentable over Anderson et al. (U.S. 5,448,735) in view of Munro (Writing DLLs for Windows using Visual Basic, part 1) further in view of Crick et al. (U.S. 5,781,797).

22. **As to claim 5**, Anderson does not explicitly teach loading the one or more code modules is performed in a reverse order of the hierarchical order. Crick teaches loading the one or more

code modules are performed in a reverse order of the hierarchical order (The driver configuration ... last load table entry; col. 5, lines 29-31). It would have been obvious to apply the teaching of Crick to the system of Anderson because it provides a method for the system to know the location/address of loaded software modules.

23. Claims 27-28 and 34-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Anderson et al. (U.S. 5,448,735) in view of Munro (Writing DLLs for Windows using Visual Basic, part 1) further in view of Mattson, Jr. (U.S. 6,317,870 B1).

24. **As to claim 27**, Anderson teaches modules are linked using pointers, or referenced in some other manner well known to those skilled in the art (col. 17, line 64 - col. 18, line 3). However, Anderson does not teach a conditional branch or jump statement includes the reference to the next one of code modules in the chain and the address in memory at which the next one of the code modules in the chain is loaded. Mattson teaches a conditional branch statement includes the reference and the address in memory at which the code module is loaded (branch (PC' + DISP'); Fig. 5A and Fig. 5B). It would have been obvious to apply the teaching of Mattson to the system of Anderson because it provides a method to move to different section of code in the program.

25. **As to claim 28**, Anderson does not teach the conditional branch or jump statement is executed when the next one of the code modules in the chain identified in the conditional branch or jump statement is executed. Mattson teaches the conditional branch or jump statement is

executed when the next one of the code modules in the chain identified in the conditional branch or jump statement is executed (branch directly to the target function; col. 5, lines 43-65). It would have been obvious to apply the teaching of Mattson to the system of Anderson because it provides a method to move to different section of code in the program.

26. **As to claim 34**, see rejection of claim 27 above.

27. **As to claim 35**, see rejection of claim 28 above.

Response to Arguments

28. Applicant's arguments filed 1/5/2004 have been fully considered but they are not persuasive.

In the remark, Applicant argued in substance that (1) Anderson does not teach "an address in memory at which the next one of the code modules in the chain is loaded is associated with the reference to the next one of the code modules in the chain", that Anderson only discloses the use of pointers to reference code modules, (2) Munro teaches away from creating a chain of code modules for execution, (3) Munro does not disclose nor suggest the disadvantages of calling DLLs from a main program, (4) the combination of Anderson and Munro would be inoperable for the intended purpose because it would fail to eliminate the overhead generally required by returning to the main program after calling each DLL, and (5) it is not obvious to apply the task organization features of Anderson to configure a hardware interface of a router.

Please see the previous rejections for the response to the above arguments.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Diem K Cao whose telephone number is (571) 272-3760. The examiner can normally be reached on Monday - Friday, 8:00AM - 3:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Any response to this action should be mailed to:
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Diem Cao

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100